Reconstructing microstructures from statistical descriptors using neural cellular automata

Paul Seibert, Alexander Raßloff, Yichi Zhang, Karl Kalina, Paul Reck, Daniel Peterseim, Markus Kästner

Angaben zur Veröffentlichung / Publication details:

Seibert, Paul, Alexander Raßloff, Yichi Zhang, Karl Kalina, Paul Reck, Daniel Peterseim, and Markus Kästner. 2024. "Reconstructing microstructures from statistical descriptors using neural cellular automata." Integrating Materials and Manufacturing Innovation 13 (1): 272–87.
https://doi.org/10.1007/s40192-023-00335-1.

Integrating Materials and Manufacturing Innovation (2024) 13:272–287
https://doi.org/10.1007/s40192-023-00335-1

TECHNICAL ARTICLE

Reconstructing Microstructures From Statistical Descriptors Using Neural Cellular Automata

Paul Seibert[1] · Alexander Raßloff[1] · Yichi Zhang[1] · Karl Kalina[1] · Paul Reck[2] · Daniel Peterseim[2] · Markus Kästner[1,3]

Received: 25 September 2023 / Accepted: 6 December 2023 / Published online: 18 January 2024

Abstract
The problem of generating microstructures of complex materials in silico has been approached from various directions including simulation, Markov, deep learning and descriptor-based approaches. This work presents a hybrid method that is inspired by all four categories and has interesting scalability properties. A neural cellular automaton is trained to evolve microstructures based on local information. Unlike most machine learning-based approaches, it does not directly require a data set of reference micrographs, but is trained from statistical microstructure descriptors that can stem from a single reference. This means that the training cost scales only with the complexity of the structure and associated descriptors. Since the size of the reconstructed structures can be set during inference, even extremely large structures can be efficiently generated. Similarly, the method is very efficient if many structures are to be reconstructed from the same descriptor for statistical evaluations. The method is formulated and discussed in detail by means of various numerical experiments, demonstrating its utility and scalability.

Keywords Microstructure · Reconstruction · Descriptor · Neural cellular automata

Introduction

The generation and analysis of random heterogeneous composite materials is a recently emerging research topic that aims at accelerating materials engineering by enabling digital workflows such as numerical simulation and inverse design [1]. Specifically, microstructure characterization and reconstruction allows to *(i)* generate many microstructure realizations from a single example, *(ii)* explore hypothetical materials by interpolating between microstructures in a morphologically meaningful manner, and *(iii)* create 3D models from 2D observations. A multitude of approaches has been developed in the last decades that is summarized in different review articles [2–4]. For the purpose of this work, the existing approaches can be broadly divided in four categories[1]—simulation, Markov random field, deep learning and descriptor-based approaches. Naturally, some algorithms in the literature can be identified as hybrid methods that fall into two or more of these categories. After discussing the main ideas of these categories and approaches in Sect. 1.1, this work presents an algorithm that bridges all four categories and exhibits some very interesting properties as described in Sect. 1.2.

Existing Approaches for Microstructure Reconstruction

Simulation-based approaches Simulating the microstructure evolution might be the most direct way. This requires to identify and to solve the physical (partial differential) equations (PDEs) that govern the process. An excellent overview is given in [2]. As an example, the Cahn–Hilliard equation describing phase separation [6] has been studied extensively [7–9]. Similarly, for granular structures, given a representative set of particles, realistic and dense packing can be achieved by simulating gravitational forces [10–12].

✉ Markus Kästner
markus.kaestner@tu-dresden.de

[1] Institute of Solid Mechanics, TU Dresden, Dresden, Germany

[2] Institute of Mathematics, University of Augsburg, Augsburg, Germany

[3] Dresden Center for Computational Materials Science, TU Dresden, Dresden, Germany

[1] Besides hybrid methods that fall into multiple categories, some exceptions like Wang tiles [5] do not clearly fall into any of the categories.

As a final, more complex example, grain formation in polycrystalline structures has been studied in depth. Simplified approaches reduce the description to vertices [13] or grain boundaries [14], whereas Monte Carlo methods [15] or cellular automata [16–18] are used to model the evolution of an entire 2D pixel field. Recently, neural cellular automata have been applied to solidification microstructure modeling [19]. Approaches based on the phase field method are probably the most developed. Thereby, the evolution of a diffuse indicator function is modeled by an additional differential equation [20–22] that can be solved, for example, in *OpenPhase* [23]. These approaches are often applied to simulate the complex microstructure morphologies that arise in additive manufacturing [24–26]. This non-exhaustive list indicates that a variety of physical processes are responsible for the formation of different material classes. Even if the relevant set of physical equations is selected, it can be challenging to perform the simulations due to numerical issues or difficulties in parameterizing the underlying constitutive models [25, 27]. This motivates the purely image-based approaches that are presented in the following.

Markov-based reconstruction As a first purely image-based method, this subsection discusses a class of reconstruction algorithms originally developed for computer graphics applications which are herein referred to as Markov-based approaches. For this purpose, it is worth noting that a microstructure can be modeled as a stationary Markov random field if the probability of finding a certain phase at a given location does not depend directly on the location, but only on the phase distribution in the local finite-size neighborhood.

This assumption of locality and stationarity motivates reconstruction algorithms that directly rely on this conditional probability to update individual pixels based on their neighbor's values. A very simple implementation inspired by texture synthesis [28] might determine individual pixel updates by scanning the reference data for the given neighborhood in order to compute a probability [29, 30]. It is worth noting that this approach is akin to the multi-point statistics method that has been developed in the Geosciences literature [31] and has been applied and improved substantially by Tahmasebi [32–35]. For a better scalability, improved algorithms precompute the probabilities for all neighborhoods and store them in efficient data structures for access during reconstruction [31, 36]. Direct sampling methods [37] as well as data structure-based alternatives are implemented in *MPSLIB* [38].

Despite a good local prediction quality, MRF-based approaches often fail to accurately reproduce long-range correlations. This behavior is related to the neighborhood size in the Markovian assumption: Capturing long-range correlations requires large neighborhood sizes, which are often unfeasible because of a disproportionately increased need for training data. Multigrid approaches [35, 39] have been shown to alleviate this issue to a certain extent. Furthermore, to condense the information to a compact model that is also able to interpolate missing neighborhood patterns from similar examples, supervised models have been trained to predict a pixel's phase given its neighborhood. In particular, decision trees [40] and neural networks [39, 41, 42] have been used for 2D and 3D [43] reconstruction. This motivates the discussion of purely deep learning-based approaches in the following subsection.

Deep learning-based reconstruction In deep learning-based methods, a generative model is fitted or trained on a sufficiently large data set of microstructures and is then used to sample new realizations of the same structure. Autoencoders [42, 44, 45] and generative adversarial networks (GANs) are typical examples that have been applied to MCR [46, 47]. For the latter, the merits of modifications like conditional GANs [48, 49], StyleGAN [50], and gradient penalty [51] have also been discussed in the context of microstructure generation. Applications to steels [52] and earth materials [53] show high image quality. Although GANs usually operate on 2D data, 3D-to-3D reconstruction can be achieved by using 3D convolutions [54, 55]. For reconstructing 3D data from 2D examples, a 3D generator has been combined with a 2D discriminator [56, 57]. As an alternative, the third dimension can be regarded as time by combining the GAN with a recurrent neural network [58]. To harness the advantage of both, autoencoders and GANs, they are sometimes combined by using the decoder simultaneously as a generator. This has proven advantageous for 2D-to-3D reconstruction [59–61] and for extremely small data sets [62].

As an alternative, machine learning methods like Bayesian approaches [63] and attention-based models [64–66] are equally applicable. Diffusion models, which have recently replaced GANs as state-of-the-art in general-purpose image generation, have also been applied to microstructure reconstruction [67, 68] and optimization [69, 70].

Much research is focused on identifying suitable model types and adapting them to microstructure reconstruction by enabling 2D-to-3D reconstruction [57, 61] making them applicable to small data sets [62] or ensuring that certain descriptor requirements are met [71, 72]. A major challenge lies in defining models with high accuracy that at the same time do not require large data sets to be trained on. These challenges motivate training-free models such as descriptor-based reconstruction, as presented in the next subsection.

Descriptor-based reconstruction The central idea behind descriptor-based reconstruction methods is to statistically quantify the microstructure morphology by means of descriptors like volume fractions and spatial n-point correlations [73]. Reconstructing a microstructure from a given set of descriptors can then be formulated as an optimization

problem directly in the space of possible microstructures. Here, the desired microstructure descriptors can be computed from a single microstructure example, making these methods very data-efficient.

One of the most well-known descriptor-based reconstruction methods is the Yeong–Torquato algorithm [73], which iteratively swaps individual pixels in the microstructure to solve the optimization problem. A detailed discussion is given in [74, 75]. This enables high flexibility, as descriptors can be replaced by new alternatives [76, 77] or higher-fidelity versions of the same descriptor [78, 79]. However, even with computationally inexpensive descriptors, the Yeong–Torquato algorithm becomes computationally challenging at high resolutions and in 3D, where billions of iterations are sometimes required for convergence [80]. A common solution is to use a multigrid scheme [81–85]. Further ideas include different-phase neighbor sampling rules [86], efficient descriptor updates [80, 87] and optimized directional weighing of correlation functions [78]. More information is given in [3].

As an alternative to the pixel-based Yeong–Torquato algorithm, the optimization problem can be formulated in a much lower-dimensional space. For this purpose, the microstructure is approximated by geometric objects that can be described by a few parameters, e.g., ellipsoidal inclusions [88–90] or Voronoi or Laguerre cells [91–93].

Independently from the microstructure representation [90], differentiable descriptors allow solving the optimization problem using a gradient-based optimizer. This idea is formulated as differentiable microstructure characterization and reconstruction (DMCR) [94, 95] and several approaches can be identified as special cases [71, 96, 97].

The Yeong–Torquato algorithm and improved versions of it, such as DMCR, have been successfully validated and applied to alloys and anisotropic metamaterials [85] sandstone [98], rock [99], chalk [100], various soils [101] and more. Some versions are publicly available in the open-source *MCRpy* package [102].

While descriptor-based approaches are very accurate and data-efficient since no training data set is required, they are computationally intensive. More specifically, since the optimization is directly carried out in the microstructure space, the memory and computational requirements grow quickly as the microstructure size increases, especially in 3D.

Hybrid reconstruction approaches The specific and unique advantages and disadvantages of all four categories of MCR approaches motivate hybrid methods that fall into multiple of these categories. Naturally, there is no sharp boundary between Markov-based and deep learning methods if a machine learning model like a neural network is used to predict individual pixels based on their neighborhood as in [39–43]. Furthermore, simulation by discretized (partial) differential equations and cellular automata resemble Markov-based methods in their locality, but are derived from physical principles and sometimes incorporate various physical quantities (e.g., temperature) beyond phase indicator functions. At the boundary between machine learning and descriptor-based methods, multiple sequential approaches use Gaussian random field-based methods [103] to initialize simulated annealing[2] [100, 104] and diffusion models [72]. Furthermore, the volume fractions [49, 58, 72, 105, 106], histograms [107] and Gram matrices [10, 108] are sometimes added to the loss function of deep learning-based methods as microstructure descriptors. *DRAGen* [109] combines an automaton-like growth process with a nucleation point optimization based on classical descriptors and allows to use machine learning models for generating input data. At the interface between machine learning and physical simulation, autoencoders [110] and diffusion models [12] have been used as particle generators followed by a gravity simulation for aggregate structures. Besides that, the literature comprises a large number of physics-informed neural network approaches that are not discussed herein.

Objectives and Contribution of this Work

This work presents a hybrid approach that is inspired by all these categories. Like in a simulation-based approach, a partial differential equation models the temporal evolution of the microstructure. It is, however, not derived from physics but learned by a neural network. Similar to the Markov-based methods, this network operates based on local information and is therefore called neural cellular automaton (NCA). This constraint of locality is relaxed not by increasing the neighborhood beyond a one pixel distance, but by introducing further hidden channels to the microstructure function that the NCA can use to encode relevant information. Finally, unlike common machine learning or Markov-based approaches, the NCA is not trained directly on image data or on a set of neighborhoods, but on a statistical descriptor. This requires the NCA to be retrained whenever the statistical descriptor changes, however, it reduces the amount of required data to a bare minimum. The input image only needs to enable the computation of a statistical descriptor; hence the NCA is applicable whenever classical training-free approaches like the Yeong–Torquato algorithm and DMCR can be used. Furthermore, the size of the training data is independent of the image size during training, which is again independent from the size of the reconstructed structure. Hence, microstructures of massive resolutions or numbers can be reconstructed with very limited additional computational effort. Furthermore, due to

[2] This is technically a hybrid method between two descriptor-based approaches.

the nature of NCA, the algorithm is inherently distributed, parallel and robust with respect to perturbations.

In summary, the central idea lies in modeling the differential equation governing the structure evolution by training neural cellular automata (NCA) on statistical descriptors. A detailed formulation is given in Sect. 2 and validated by various numerical experiments in Sect. 3. A conclusion is drawn in Sect. 4.

Neural Cellular Automata for Descriptor-Based Microstructure Reconstruction

Based on the work of Mordvintsev et al. [111], the formulation of general neural cellular automata (NCA) is summarized in Sect. 2.1. The main idea of the present work to train NCA by arbitrary descriptors is described in Sect. 2.2. Finally, the implementation is discussed in Sect. 2.3.

Formulation of Neural Cellular Automata

The general idea behind a cellular automaton is to iteratively update individual pixels based on the direct neighbors. In the work of Mordvintsev et al. [111], this information source is further restricted. The neighboring pixel values are not passed directly to the cellular automaton. Instead, they are used to compute a discrete approximation to the gradient and curvature, which are then passed to the cellular automaton. Denoting $\boldsymbol{x} \in \mathcal{D}$ as a position vector in the microstructure domain $\mathcal{D} \subset \mathbb{R}^2$ and $t \in \mathcal{T} = \{t \in \mathbb{R} \mid 0 \leq t \leq t^{\text{end}}\}$ as time, the evolution of the microstructure $m(\boldsymbol{x}, t)$ can be written as a partial differential equation

$$\frac{\partial m(\boldsymbol{x},t)}{\partial t} = f_{\boldsymbol{\theta}}\left(m(\boldsymbol{x},t), \nabla_{\boldsymbol{x}} m(\boldsymbol{x},t), \nabla_{\boldsymbol{x}}^2 m(\boldsymbol{x},t)\right) , \tag{1}$$

where $f_{\boldsymbol{\theta}}$ is the cellular automaton which maps the value, gradient and curvature of the microstructure function to its temporal derivative. To be more specific, $\nabla_x(\cdot)$ and $\nabla_x^2(\cdot)$ denote the gradient and Laplace operator, respectively. Furthermore, m takes real values within the arbitrarily chosen bounds

$$0 \leq m(\boldsymbol{x},t) \leq 1 \quad \forall\, \boldsymbol{x} \in \mathcal{D},\ t \in \mathcal{T}. \tag{2}$$

A promising extension to multi-phase materials might be to declare m as a vector-valued function, where each entry represents the indicator function of a distinct phase. In this context, the mutually exclusive nature of phase indicator functions should be considered, for example by a penalty factor. In a neural cellular automaton specifically, a neural network is chosen as $f_{\boldsymbol{\theta}}$ in Eq. 1, where $\boldsymbol{\theta}$ denotes the parameter vector.

In other words, the NCA defines partial differential equation (PDE),[3] that needs to be discretized and solved in order to generate a microstructure. An explicit Euler scheme is chosen as a time stepping scheme

$$\frac{m_{n_t+1} - m_{n_t}}{\Delta t} = f_{\boldsymbol{\theta}}\left(m_{n_t}, \nabla_{\boldsymbol{x}} m_{n_t}, \nabla_{\boldsymbol{x}}^2 m_{n_t}\right) , \tag{3}$$

where the current solution at time step n_t defines the update for the next time step $n_t + 1$. The dependence on $\boldsymbol{x}$ and t is dropped for the sake of brevity. The space is naturally discretized on an equidistant grid of pixel values, where $\nabla_x(\cdot)$ and $\nabla_x^2(\cdot)$ are approximated by a Sobel and Laplace filter, respectively. Based on this discretization, the relation between the current solution, its spatial derivatives and its temporal evolution, i.e., the PDE itself, is learned by the NCA.

Given the inability of Markov-based approaches with small neighborhood sizes to accurately capture long-range correlations, it should be clear that the extremely limited local information is not sufficient to train a good NCA. For this reason, the augmented microstructure function $\boldsymbol{m'}(\boldsymbol{x}, t)$ is introduced which maps a spatial position $\boldsymbol{x}$ at time t to an n-dimensional vector. The first entry of the vector contains the normal microstructure function $m(\boldsymbol{x}, t)$ and is the only entry that affects the training. The idea behind the other entries is that the NCA can choose to allocate any quantity that is useful for passing information and increasing the image quality. As an example, for an equiaxed grain microstructure, one channel might contain the distance to the next grain boundary. With this, the temporal evolution reads

$$\frac{\partial \boldsymbol{m'}(\boldsymbol{x},t)}{\partial t} = f_{\boldsymbol{\theta}}\left(\boldsymbol{m'}(\boldsymbol{x},t), \nabla_{\boldsymbol{x}} \boldsymbol{m'}(\boldsymbol{x},t), \nabla_{\boldsymbol{x}}^2 \boldsymbol{m'}(\boldsymbol{x},t)\right) \tag{4}$$

For reconstructing a microstructure from the trained NCA, $\boldsymbol{m'}(\boldsymbol{x}, 0)$ is initialized by zeros and the system evolves freely. Although the learned automaton itself is deterministic, the random choice of pixels to update in each iteration makes the output stochastic. Hence, the trained NCA can be used for efficiently generating many realizations of a random heterogeneous medium that are different but statistically equivalent.

Training the Model from Microstructure Descriptors

The function $f_{\boldsymbol{\theta}}$ is learned by a small neural network with two layers as shown in Fig. 1: *First*, an initial solution $\boldsymbol{m'}(\boldsymbol{x}, 0) = \boldsymbol{0}\ \forall\, \boldsymbol{x}$ is chosen. *Secondly*, $\boldsymbol{m'}(\boldsymbol{x}, t)$ develops according to Eq. 4 for a randomly chosen number of

[3] To be precise, the NCA defines a PDE *system* as explained later in the document.

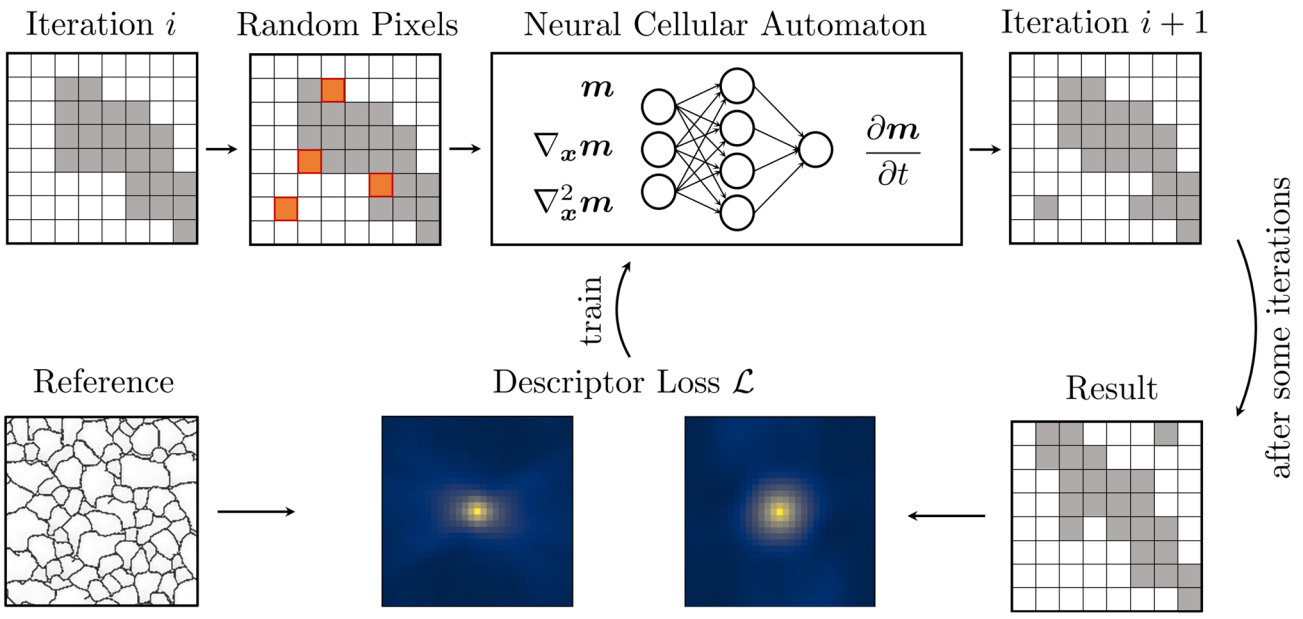


Fig. 1 Training procedure for a neural cellular automaton (NCA): In every iteration i, random pixel locations are chosen where the gradient and curvature is computed numerically. Together with the pixel value, these quantities are given to the NCA to predict a pixel update. After some time increments, the result is compared to the reference to train the NCA. This comparison is only carried out in terms of statistical descriptors

time steps. As a regularization and as a measure to break symmetry, asynchronous updates are chosen, whereby in every time step, a given percentage of cells is chosen at random and only those develop. This fraction is referred to as *fire rate*. The bounds given in Eq. 2 are enforced by clipping. *Thirdly*, a loss function $\mathcal{L}$ is computed on the final result $m^{\text{end}} = m(\boldsymbol{x}, t^{\text{end}})$. The choice of $\mathcal{L}$ is discussed later. Note that only m^{end}, i.e., the first component of $\boldsymbol{m'}^{\text{end}}$, contributes to $\mathcal{L}$. *Finally*, the gradient $\partial\mathcal{L}/\partial\boldsymbol{\theta}$ of the loss function with respect to the NCA parameters is computed by conventional backpropagation and used to update $\boldsymbol{\theta}$. Note that this limits the number of timesteps during training for numerical reasons.

The formulation of the loss function depends on the area of application of the NCA. After initially using a pixel-wise Euclidean norm error in the RGB space for general-purpose image generation [111], Mordvintsev et al. [112] found that a Gram matrix-based style loss [113] enable NCAs to be applied to texture synthesis [112].

The novelty in the present work lies in realizing that any of the known statistical descriptors can be used, as long as they can be differentiated with respect to the microstructure field. This concept is discussed extensively in [94] and summarized in the following. In practice, the statistical descriptors are mere functions that take the pixel representation of the discretized structure as input. Hence, first and foremost, the descriptors need to be defined for any pixel values within the allowed bounds in Eq. 2. This stands in contrast to the Yeong–Torquato algorithm, where the microstructure function takes one and only one phase ID as value at each pixel. Moreover, the descriptor functions must be continuous to imply differentiability. To achieve this, sharp thresholds in a descriptor definition may need to be replaced by a smooth transition for defining a differentiable approximation. Finally, the descriptors should be sensitive to changes in the microstructure function. This requirement is needed for deriving microstructure updates from descriptor gradients. Concretely, it means that the gradient of the descriptor with respect to the microstructure parametrization should not only exist, but also be nonzero as often as possible.

The loss is thus formulated as a mean-squared error (MSE) in the descriptor space

$$\mathcal{L} = \|\boldsymbol{D}(m^{\text{end}}) - \boldsymbol{D}^{\text{des}}\|_{\text{MSE}} \,, \tag{5}$$

where $\boldsymbol{D}$ denotes a statistical descriptor or a weighted concatenation of multiple descriptors that is computed on the reconstruction result, while $\boldsymbol{D}^{\text{des}}$ denotes the desired value computed from the reference structure. Because m^{end} results from the temporal evolution of $f_{\boldsymbol{\theta}}$, it depends on the parameter vector $\boldsymbol{\theta}$ of the NCA. The central idea is training the NCA by gradient-based optimization of $\boldsymbol{\theta}$ to minimize Eq. 5, whereby arbitrary descriptors can be incorporated. While the Gram matrices used in [112] can be interpreted as a

statistical descriptor, the spatial two- and three-point correlations are more common in microstructure reconstruction. The idea of using high-dimensional, differentiable descriptors for direct microstructure reconstruction is given in [94], where an efficient and differentiable formulation of the three-point correlations is given. As another example, a differentiable approximation to lineal path function is presented in [102] and a descriptor based on a hierarchical wavelet transform is given in [114]. All these descriptors are implemented in *MCRpy* [102].

Implementation

The implementation of a descriptor-based NCA for microstructure reconstruction is carried out based on the code for NCA texture synthesis [112] and the differentiable descriptors available in *MCRpy* [102]. The former code is adapted to only a single non-hidden dimension $m(\boldsymbol{x})$ as opposed to three RGB channels. Then, *MCRpy* is used to define a loss, where different descriptors such as Gram Matrices G [71], correlations S [74], variation $\mathcal{V}$ [95] and volume fraction φ can be combined and weighed in a single loss function in a flexible manner. More information on these descriptors is given in [102]. *MCRpy* makes use of the automatic differentiation in *TensorFlow* to compute the gradient $\partial\mathcal{L}/\partial m$. Then, m is backpropagated through time to compute $\partial m/\partial\theta$ and consequently $\partial\mathcal{L}/\partial\theta$.

Finally, a hyperparameter study is carried out on a number of structures. A 12-dimensional microstructure representation (i.e., 11 hidden channels) is chosen. Hence, the NCA has 12 output and 48 input dimensions. With a single hidden layer of 120 neurons, the network amounts to a total of 7332 parameters. Further hyperparameters like the number of time steps are summarized in Table 1.

In order to visually compare the results of descriptor-based NCA with other methods from the literature, three open-source codes are selected from GitHub. To represent Markov-based methods, a patch-based texture synthesis[4] algorithm based on [115, 116] and a pixel-based, multi-resolution texture synthesis[5] algorithm based on [117, 118] are chosen. Furthermore, *MCRpy* [102] implements differentiable microstructure characterization and reconstruction (DMCR) [94, 95]. While *MCRpy* is provided by previous works of the authors, the former two methods are coded and provided by Anastasia Opara. The authors greatly acknowledge this effort and appreciate the will to share software.

[4] https://github.com/anopara/texture-synthesis-nonparametric-sampling.

[5] https://github.com/anopara/multi-resolution-texture-synthesis.

Table 1 Hyperparameters chosen in the present work

Parameter	Value
Hidden layer size	120
Non-hidden channels	1
Hidden channels	11
Activation function	ReLU
Fire rate	0.5
Batch size	4
Checkpointing[1] pool size	1024
Learning rate	$2 \cdot 10^{-3}$
Rollout length probability	$\mathcal{U}(32, 64)$
Gradient normalization	True
Overflow loss coeff	10^4
Descriptors	$S, G, \mathcal{V}$
Descriptor weights	1, 1, 100

Therein, the fire rate denotes the fraction of randomly chosen pixels to update in each iteration. The descriptors are weighed before combining them to a loss function to compensate for the small numerical value of the variation. The weights are given in the same order as the corresponding descriptors. For the remaining parameters, the reader is referred to the work by Mordvintsev et al. [112].[1]This technique is outlined in [112] and aims to prevent the structures from decaying for long simulation times

Numerical Experiments

The microstructure evolution and the range of applicability are investigated in Sect. 3.1. These results are then compared to the literature in Sect. 3.2. Finally, the scalability of descriptor-based NCA is demonstrated in Sect. 3.3. All numerical experiments are carried out on a laptop with a 12 th Gen Intel(R) Core(TM) i7-12800 H CPU at 2.40 GHz and an Nvidia A2000 GPU with 4 GB VRAM.

Microstructure Evolution and Diversity

Figure 2 shows reconstructions from different real materials taken from [71]. The reconstructed structures are *not* thresholded or post-processed in any other way, but the trained NCA itself drives the solution to an almost binary structure. It can be seen that descriptor-based NCA are applicable to a wide variety of fundamentally different structures, ranging from relatively noise-free examples like the grain boundary structure and the ceramics to the more noisy sandstone. Some limitations can also be seen. As a first limitation, although the grain boundary character in the alloy is captured relatively well, not all lines are connected as in the reference case. In order to use the results for downstream tasks like numerical simulations, a post-processing algorithm is first needed to close the gaps or eliminate unnecessary line segments. Alternatively, it might be worth investigating if

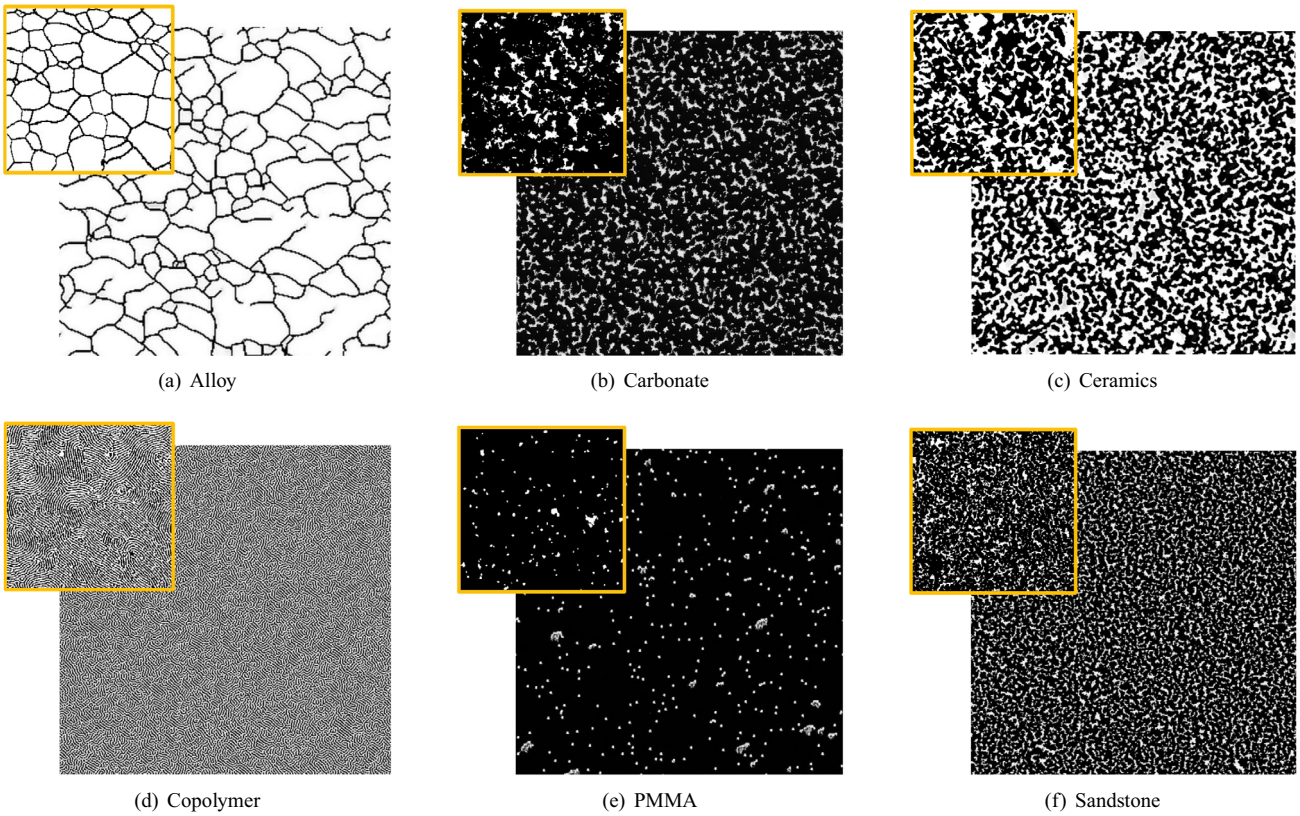
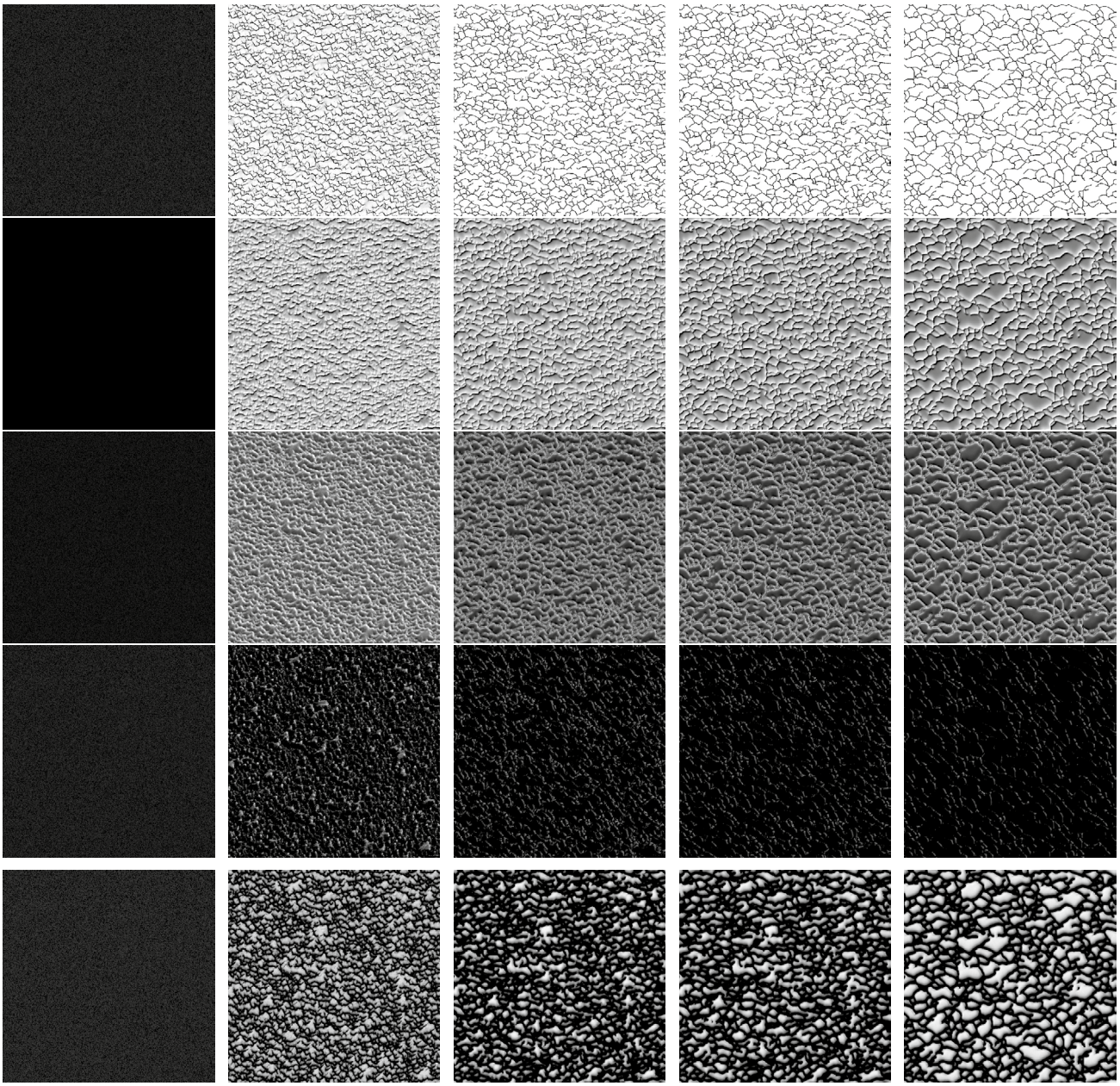


Fig. 2 Reconstructions from various real materials. No thresholding or any other type of post-processing is applied. The original samples are given in the top left corner and are taken from [71], where they are released under the Creative Commons license [119]

a different choice of descriptor can be used in order to better quantify the connectivity information. Although this approach is arguably more elegant, its difficulty lies in the requirement that the descriptor should be differentiable with respect to the microstructure. As a second limitation, it can be seen that the fingerprint-like structure of the copolymer is not adequately represented. Although the NCA successfully creates individual sub-regions with parallel lines, these regions are not sufficiently large and the lines do not exhibit smooth curves as in the reference. It is presently unclear to the authors how this issue can be addressed. As a third limitation, it is noted that the probability distribution of pixel values does not exactly match the original structures. Especially in the carbonate and PMMA, it can be seen that the white phase is reconstructed in bright gray color. Similar to the first limitation, the authors assume that a post-processing algorithm or a suitable descriptor should be sufficient to address this issue.

To provide a better understanding of the generation method, the temporal evolution of the microstructure as well as the first four hidden dimensions is plotted in Fig. 3. All fields are initialized by zero (black) and the structure slowly emerges. Different hidden channels take different roles in representing structural features. For example, the first hidden channel (second row) might be interpreted as a local vertical coordinate in each grain. In contrast, the fourth hidden channel (last row) contains a thickened version of the gain boundaries. Interestingly, the third hidden channel (second to last row) can be interpreted in different ways. It might be used as a type of residuum, since its norm decreases as the reconstruction converges. As an alternative, it might act as a marker for specific features like triple junctions. It can be concluded that different channels take different roles, although a direct interpretation is neither possible, nor necessary.

It is demonstrated in the works of Mordvintsev et al. [111, 112] that the NCA-based generation process is often robust with respect to perturbations. To test whether this trend is transferred to descriptor-based NCA for microstructure reconstruction, two numerical experiments are carried out. After the generation process has converged a good solution, the structure is perturbed by setting all values within a circular radius to 0.5. This is applied to all channels in Fig. 4 and only to the non-hidden dimension in Fig. 5. It can be seen that the structure only recovers in the latter case. Besides stressing the key role of the hidden channels, this indicates that the robustness of NCA



(a) $t = 0$ (b) $t = 30$ (c) $t = 60$ (d) $t = 90$ (e) $t = 300$

Fig. 3 The evolution of the alloy microstructure over time t. The first channel is plotted in the top row and the first four hidden channels are given below. It can be seen that each hidden channel acts as a distinct feature map

is only partially observed in descriptor-based NCA for microstructure reconstruction.

Comparison to Literature

In order to compare descriptor-based NCA reconstruction results to the literature, two Markov- and one descriptor-based approach is chosen.[6] Figure 6 shows the results, where only three material classes are selected for

[6] The authors were not able to apply deep learning-based algorithms, since only a single reference sample is available. Similarly, simulation-based approaches are omitted because they require a detailed analysis and modeling of the materials' underlying physics that exceeds the scope of this work.

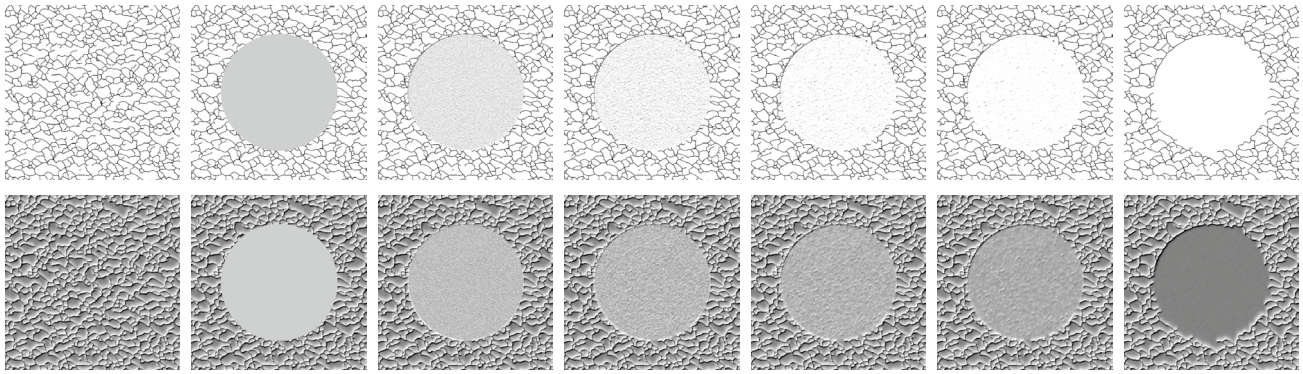


Fig. 4 The role of the hidden channels is illustrated by perturbing the microstructure evolution at time $t = t'$. All pixel values within a given radius are set to 0.5. In the presented case, all channels are perturbed, whereas in Fig. 5, the hidden channels remain intact. Only the microstructure (top) and the first hidden channel (bottom) are plotted for brevity. Unlike in Fig. 5, the structure does not recover

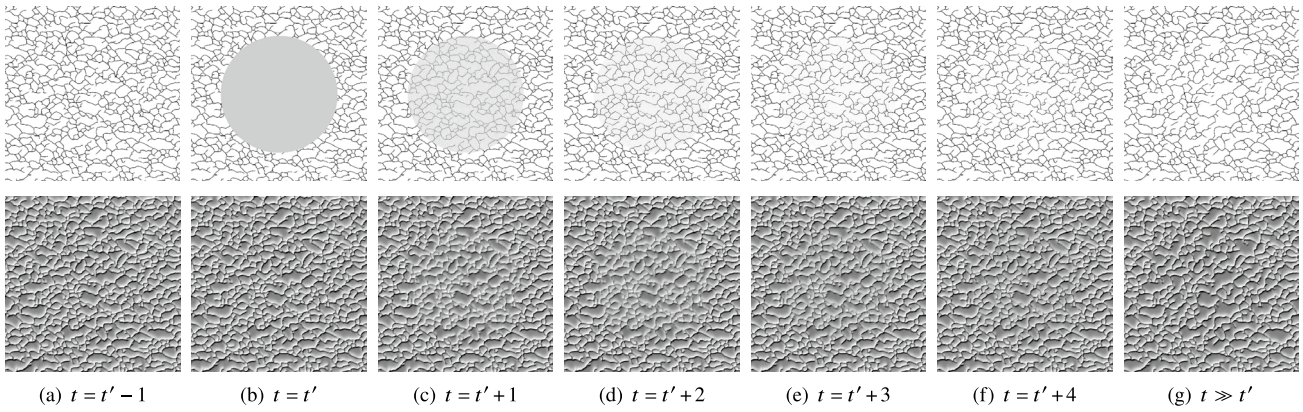


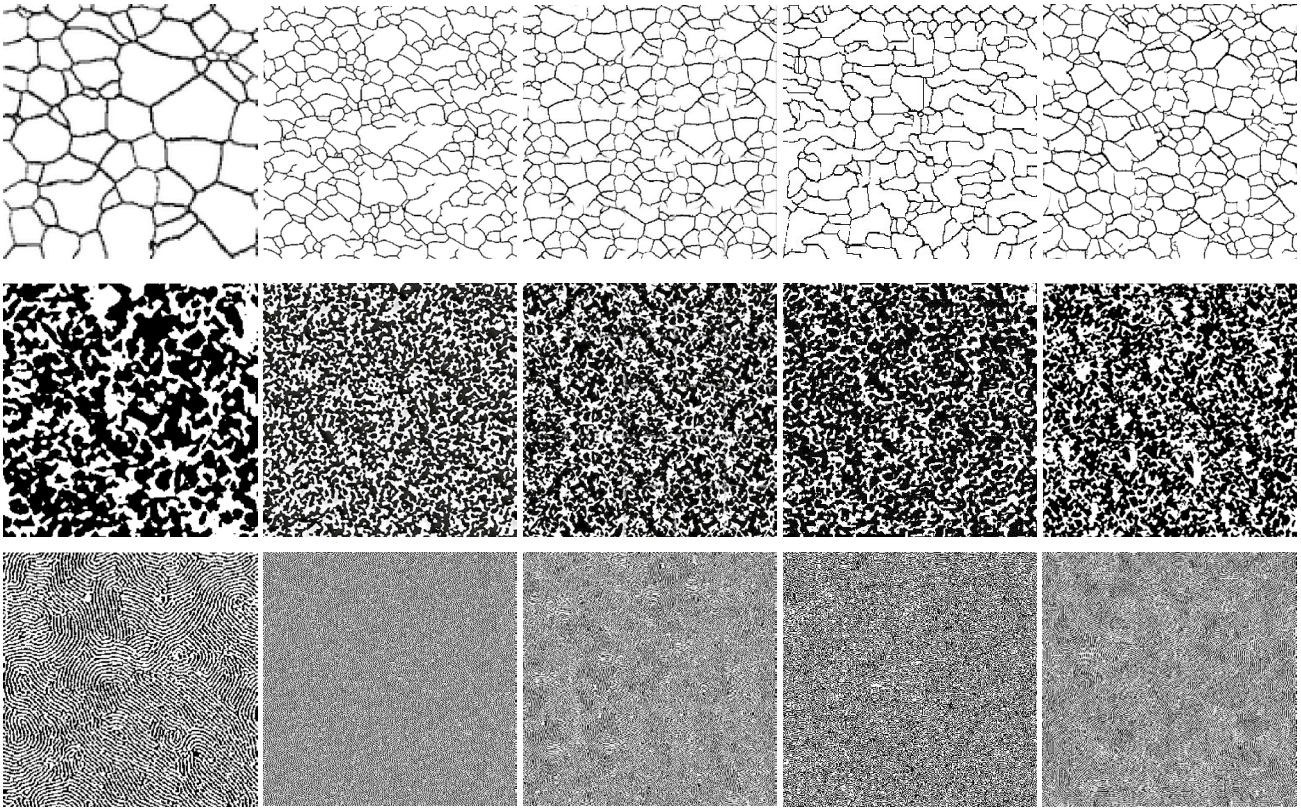
Fig. 5 The role of the hidden channels is illustrated by perturbing the microstructure evolution at time $t = t'$. All pixel values within a given radius are set to 0.5. In the presented case, the hidden channels remain intact, whereas in Fig. 4, all channels are perturbed. Only the microstructure (top) and the first hidden channel (bottom) are plotted for brevity. Unlike in Fig. 4, the structure recovers, albeit to a different solution

the sake of brevity. At a first glance, all methods produce high-quality results. Patch-based texture synthesis, however, does not produce new structural features, but copies paths from the original structure to different locations in the target image. The patch boundaries can be distinguished upon closer inspection. As a pixel-based approach, multi-resolution texture synthesis does not suffer from this phenomenon. However, especially in the alloy, strange features like completely vertical grain boundaries can be observed and the structure coincidentally repeats itself in the top right corner. Furthermore, the highly complex fingerprint-like copolymer structure is not captured adequately. Finally, DMCR as a descriptor-based method[7] produces good results for all considered materials. While the alloy and ceramics are similarly well reconstructed as with the descriptor-based NCA, DMCR produces visually superior results for the copolymer. It can be concluded that the result quality of NCA outperforms standard Markov-based techniques and almost reaches that of direct descriptor-based optimization. The advantage, with respect to the latter lies in the performance and scalability as discussed in the following.

Performance and Scalability

An objective assessment of the reconstruction results in terms of microstructure descriptors is paramount to evaluating the accuracy of any reconstruction algorithm. In this context, it should be mentioned that the presented method is only partially descriptor-based since the descriptors are

[7] The Yeong–Torquato algorithm can be expected to yield equally good or even better results, however, at a significantly higher computational cost.



(a) Reference (b) NCA (c) PBTS (d) MRTS (e) DMCR

Fig. 6 Comparison of three selected materials reconstructions with methods from the literature. Patch-based texture synthesis (PBTS) and multi-resolution texture synthesis are Markov-based approaches, while differentiable microstructure characterization and reconstruction (DMCR) is descriptor-based. The reconstructed structures are two times larger than the reference in each direction. More information is given in Sect. 2.3

used during training, but not during sampling. For this reason, independent realizations of the material exhibit random deviations from the target descriptor. Naturally, these fluctuations are expected to decrease as the microstructure size increases.

This is shown in Fig. 7, where the error

$$\mathcal{E}_D = \|\boldsymbol{D}(m^{\text{end}}) - \boldsymbol{D}^{\text{des}}\|_{\text{MSE}} \tag{6}$$

between a descriptor $\boldsymbol{D}(m^{\text{end}})$ and its desired value $\boldsymbol{D}^{\text{des}}$ from the reference structure is defined as a mean-squared error (MSE). Note that the only difference between $\mathcal{E}_D$ and the loss $\mathcal{L}$ defined in Eq. 5 is that the former measures individual descriptors, whereas the latter is based on a weighted concatenation of multiple descriptors. For all tested descriptors, the error converges to a value which is consistently lower for the proposed loss model than for the reference NCA-based texture synthesis method by Mordvintsev et al. [112]. It should be noted that the descriptor errors do not converge to zero as the resolution increases, but rather to a value that depends on the training quality. It is observed that longer training and training by larger samples reduces this value (not shown here).

An interesting aspect of NCA is that the image sizes during training and sampling are independent. This is favorable because the sampling is relatively inexpensive and scales favorably with the image size compared to other methods. To demonstrate this, Fig. 8 shows a reconstruction example where the resolution is chosen such that the sampling takes as long as the training.[8] Three different zoom levels of the same structure are shown for visualization purposes. Without any multigrid procedures, such large reconstructions are very challenging with classical descriptor-based methods.

Generally, the computational cost of sampling a microstructure scales linearly in the number of pixels, because

[8] Because the utilized VRAM is not sufficient for the large reconstruction, it is conducted on the CPU only, whereas the training occurs on the GPU. If a similar comparison was made on identical hardware, significantly larger structures could be reconstructed. Regardless, in the authors' opinion, the presented results demonstrate the scalability sufficiently well.

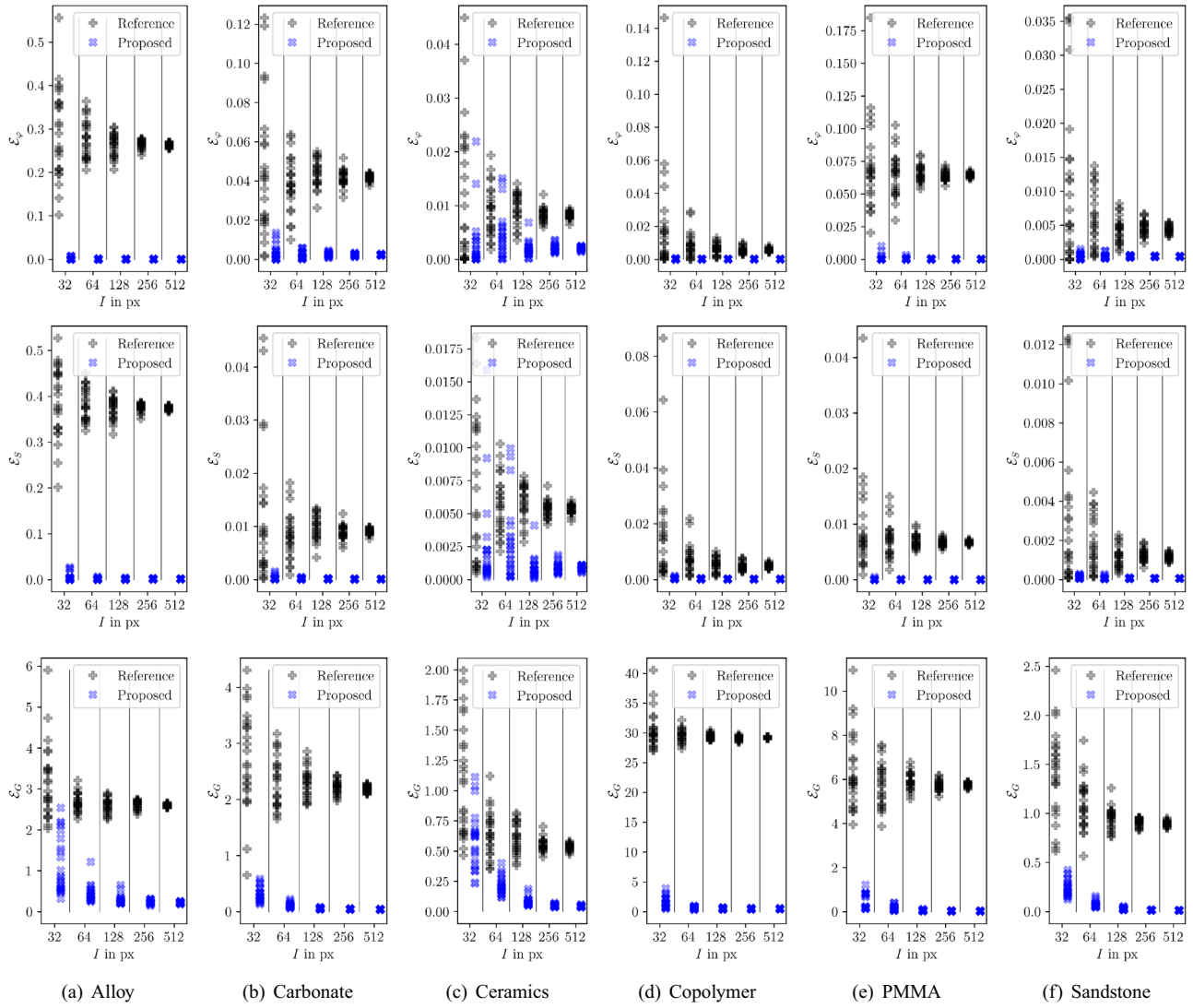


Fig. 7 Influence of the loss function on the descriptor errors. The volume fractions φ (top), spatial correlations S (middle) and Gram matrices G (bottom) are compared for different materials (left to right) with 25 realizations per resolution. The resolutions are powers of two and an offset to the left (reference) and right (proposed) is applied only for visualization purposes. If can be seen that regardless of the model, material and descriptor, the variance of the descriptor error over different realizations decreases as the sample size increases. The proposed model consistently outperforms the reference [112]. For some structures like the ally (a), the reference model fails to converge, leading to massive discrepancies, whereas for the ceramics (c) the differences are relatively small

pixel updates are computed independently by the NCA. A comparison with the 2D DMCR algorithm [94] in *MCRpy* [102] is given in Fig. 9. Both methods scale linearly. Sampling from a trained NCA is much faster than reconstructing by DMCR, and the computational cost grows more slowly. This is because the expensive evaluation of microstructure descriptors and iterative optimization are moved to the training stage. If the training is added to the computational cost of the NCA, they are slower for the considered microstructure sizes. As a conclusion, the expensive training phase of an NCA is compensated if large or many microstructures are reconstructed. Particularly, the latter might speed up a potential future extension for 2D-to-3D reconstruction. Furthermore, unlike with DMCR [94, 95], the sampling can be trivially parallelized because updates are based only on local information.

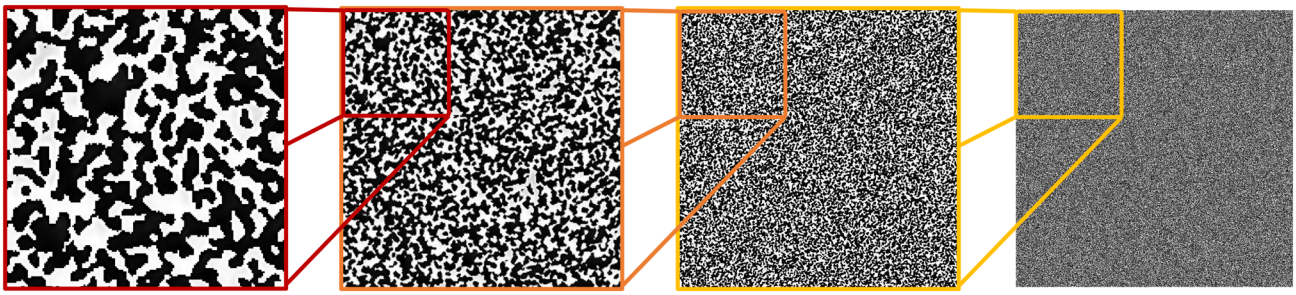


Fig. 8 A reconstruction of the ceramics microstructure (Fig. 2c) with 4096^2 pixels shows the scalability of the method. Three different zoom levels of the same structure are shown for visualization purposes

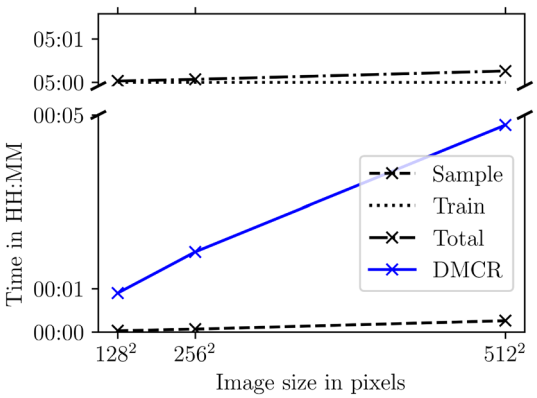


Fig. 9 Scalability comparison of descriptor-based NCA with the 2D DMCR algorithm [94] in *MCRpy* [102]. Both scale linearly in the number of pixels, which scales quadratically in the side length. The constant offset of the NCA cost due to training is compensated by the much faster sampling, which also grows at a slower rate

Conclusion

A neural cellular automaton (NCA)-based algorithm for microstructure reconstruction is presented. The microstructure evolution is modeled as a partial differential equation which is learned by a small neural network, the NCA. Despite the purely local information in the NCA, long-range correlations are incorporated by introducing hidden dimensions to the microstructure function which can be used to communicate information. Unlike with previous approaches, this network is not trained on image data but on statistical microstructure descriptors. Thus, the method incorporates ideas from four different families of microstructure generation approaches, namely simulation, Markov, deep learning and descriptor-based methods, which are all briefly reviewed. The method is formulated, implemented and validated by a number of 2D numerical experiments.

Compared to other microstructure reconstruction approaches, descriptor-based NCAs have a unique set of advantages. The neural network in the NCA enables the evolution of highly complex morphologies in a PDE-like manner without knowledge of the governing physical equations and the material parameters. It can be controlled by statistical descriptors. However, the sampling of structures from a trained NCA is based only on local information. This self-assembling nature of the algorithm makes it an inherently distributed algorithm and therefore trivial to parallelize. The random selection of the pixels to be updated makes the method robust with respect to random perturbations, as long as not all channels are affected. Finally, the method scales very favorably as arbitrarily resolved structures can be sampled.

In future work, the main challenge lies in enabling 3D reconstruction based on 2D or 3D reference data. The difficulty therein lies in the memory and computational cost of the training procedure. Specifically, the backpropagation through all time steps is a bottleneck that is manageable in 2D, but becomes unfeasible in 3D on current hardware. As an alternative, algorithms might be developed that apply a trained 2D NCA to sections of a 3D structure. Finally, applications to polycrystalline structures motivate an extension of the NCA to orientation information. Besides introducing a vector-valued microstructure function in analogy to the discussed multi-phase extension, this requires the definition of adequate statistical descriptors. To be specific, the descriptors need to be symmetrized as given by the crystal structure and differentiable, even across the boundaries of the fundamental zone. Once developed, these descriptors also enable extensions of optimization-based microstructure reconstruction algorithms like DMCR.

Acknowledgements The authors thank Anastasia Opara for providing good implementations of texture synthesis algorithms to the community. The groups of M. Kästner and D. Peterseim thank the German Research Foundation DFG which supported this work under Grant numbers KA 3309/18-1 and PE 2143/7-1, respectively.

Author Contributions PS contributed to conceptualization, data curation, formal analysis, investigation, methodology, software, supervision, validation, visualization, writing—original draft preparation, and

writing—review and editing. AR and PR performed conceptualization and writing—review and editing. YZ contributed to software and writing—review and editing. KK performed conceptualization, formal analysis, and writing—review and editing. DP was involved in conceptualization, funding acquisition, and writing—review and editing. MK contributed to conceptualization, funding acquisition, resources, supervision, and writing—review and editing.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data availability The code is made available upon reasonable request. The data are taken from the literature [71], where it is released under the Creative Commons license.

Declarations

Conflict of interest The authors declare no competing interests.

References

1. Chen W, Iyer A, Bostanabad R (2022) Data-centric design of microstructural materials systems, engineering. p S209580992200056X. https://doi.org/10.1016/j.eng.2021.05.022. https://linkinghub.elsevier.com/retrieve/pii/S2095809922000056X
2. Bargmann S, Klusemann B, Markmann J, Schnabel JE, Schneider K, Soyarslan C, Wilmers J (2018) Generation of 3D representative volume elements for heterogeneous materials: a review. Progress Mater Sci 96:322
3. Bostanabad R, Zhang Y, Li X, Kearney T, Brinson LC, Apley DW, Liu WK, Chen W (2018) Computational microstructure characterization and reconstruction: review of the state-of-the-art techniques. Progress Mater Sci 95:1. https://doi.org/10.1016/j.pmatsci.2018.01.005
4. Sahimi M, Tahmasebi P (2021) Reconstruction, optimization, and design of heterogeneous materials and media: basic principles, computational algorithms, and applications. Phys Rep 939:1
5. Doškář M, Zeman J, Krysl P, Novák J (2021) Microstructure-informed reduced modes synthesized with Wang tiles and the generalized finite element method. Comput Mech 68(2):233
6. Cahn JW, Hilliard JE (1958) Free energy of a nonuniform system. I. Interfacial free energy. J Chem Phys 28(2):258
7. Wight CL, Zhao J (2021) Solving Allen-Cahn and Cahn-Hilliard equations using the adaptive physics informed neural networks. Commun Comput Phys 29:930
8. Prakasha D, Veeresha P, Baskonus H (2019) Two novel computational techniques for fractional Gardner and Cahn-Hilliard. Comp Math Methods 1(2):1021
9. Risthaus L, Schneider M (2022) Solving phase-field models in the tensor train format to generate microstructures of bicontinuous composites. Appl Numer Math 178:262
10. Zhao X, Wu X, Wang L, Hou P, Li Q, Zhang Y, Yang B (2022) Three-dimensional microstructural image synthesis from 2D backscattered electron image of cement paste. arXiv:2204.01645 [cs, eess]
11. Winkler P, Sadaghiani M, Jentsch H, Witt K (2014) Granular packing generation using DEM-modified force-biased-algorithm. Scour and Erosion. CRC Press, Boca Raton, pp 345–349
12. Vlassis NN, Sun W, Alshibli KA, Regueiro RA (2023) Synthesizing realistic sand assemblies with denoising diffusion in latent space, arXiv:2306.0441 [cs:CE]
13. Kawasaki K, Nagai T, Nakashima K (1989) Vertex models for two-dimensional grain growth. Philos Mag B 60(3):399
14. Brakke KA (1992) The surface evolver. Exp Math 1(2):141
15. Anderson MP, Grest GS, Srolovitz DJ (1989) Computer simulation of normal grain growth in three dimensions. Philos Mag B 59(3):293
16. Janssens KGF (2010) An introductory review of cellular automata modeling of moving grain boundaries in polycrystalline materials. Math Comput Simul (MATCOM) 80(7):1361
17. Zhan X, Wei Y, Dong Z (2008) Cellular automaton simulation of grain growth with different orientation angles during solidification process. J Mater Process Technol 208(1):1
18. Ghumman UF, Fang L, Wagner GJ, Chen W (2023) Calibration of cellular automaton model for microstructure prediction in additive manufacturing using dissimilarity score. J Manuf Sci Eng 145:061002. https://doi.org/10.1115/1.4056690
19. Tang J, Kumar S, De Lorenzis L, Hosseini E (2023) Neural cellular automata for solidification microstructure modelling. Comput Methods Appl Mech Eng 414:116197. https://doi.org/10.1016/j.cma.2023.116197
20. Boettinger WJ, Warren JA, Beckermann C, Karma A (2002) Phase-field simulation of solidification. Ann Rev Mater Res 32(1):163
21. Tourret D, Liu H, Lorca J (2022) Phase-field modeling of microstructure evolution: recent applications, perspectives and challenges. Progress Mater Sci 123:100810
22. Skogvoll V, Salvalaglio M, Angheluta L (2022) Hydrodynamic phase field crystal approach to interfaces, dislocations, and multigrain networks. Model Simul Mater Sci Eng 30:084002
23. Tegeler M, Shchyglo O, Kamachali RD, Monas A, Steinbach I, Sutmann G (2017) Parallel multiphase field simulations with OpenPhase. Comput Phys Commun 215:173
24. Tan JHK, Sing SL, Yeong WY (2020) Microstructure modelling for metallic additive manufacturing: a review. Virtual Phys Prototyp 15(1):87
25. Körner C, Markl M, Koepf JA (2020) Modeling and simulation of microstructure evolution for additive manufacturing of metals: a critical review. Metall Mater Trans A 51(10):4970
26. Li J, Zhou X, Brochu M, Provatas N, Zhao YF (2020) Solidification microstructure simulation of Ti-6Al-4V in metal additive manufacturing: a review. Additive Manuf 31:100989
27. Yan W, Lin S, Kafka OL, Lian Y, Yu C, Liu Z, Yan J, Wolff S, Wu H, Ndip-Agbor E, Mozaffar M, Ehmann K, Cao J, Wagner GJ, Liu WK (2018) Data-driven multi-scale multi-physics models to derive process–structure–property relationships for additive manufacturing. Comput Mech 61(5):521
28. Wei LY, Lefebvre S, Kwatra V, Turk G (2009) State of the art in example-based texture synthesis, Eurographics. State of the Art Report, EG-STAR, pp 93–117
29. Sundararaghavan V (2014) Reconstruction of three-dimensional anisotropic microstructures from two-dimensional micrographs imaged on orthogonal planes. Integr Mater Manuf Innov 3(1):240

30. Liu X, Shapiro V (2015) Random heterogeneous materials via texture synthesis. Comput Mater Sci 99:177
31. Strebelle S (2002) Conditional simulation of complex geological structures using multiple-point statistics. Math Geol 34:21
32. Tahmasebi P, Sahimi M (2013) Cross-correlation function for accurate reconstruction of heterogeneous media. Phys Rev Lett 110(7):078002
33. Tahmasebi P, Sahimi M (2016) Enhancing multiple-point geostatistical modeling: 1 graph theory and pattern adjustment: enhancing geostatistical modeling. Water Resour Res 52(3):2074
34. Tahmasebi P, Sahimi M (2016) Enhancing multiple-point geostatistical modeling: 2. Iterative simulation and multiple distance function: enhancing geostatistical modeling. Water Resour Res 52(3):2099
35. Tahmasebi P, Sahimi M (2018) A stochastic multiscale algorithm for modeling complex granular materials. Granul Matter 20(3):45
36. Straubhaar J, Renard P, Mariethoz G, Froidevaux R, Besson O (2011) An improved parallel multiple-point algorithm using a list approach. Math Geosci 43(3):305
37. Mariethoz G, Renard P, Straubhaar J (2010) The direct sampling method to perform multiple-point geostatistical simulations. Water Resour Res 45:85. https://doi.org/10.1029/2008WR007621
38. Hansen TM, Vu LT, Bach T (2016) MPSLIB: a C++ class for sequential simulation of multiple-point statistical models. SoftwareX 5:127
39. Fu J, Wang M, Xiao D, Zhong S, Ge X, Wu M, Evans B (2023) Hierarchical reconstruction of 3D well-connected porous media from 2D exemplars using statistics-informed neural network. Comput Methods Appl Mech Eng 410:116049
40. Bostanabad R, Bui AT, Xie W, Apley DW, Chen W (2016) Stochastic microstructure characterization and reconstruction via supervised learning. Acta Materialia 103:89. https://doi.org/10.1016/j.actamat.2015.09.044
41. Latka K, Doskar M, Zeman J (2021) Microstructure reconstruction via artificial neural networks: a combination of causal and non-causal approach. Acta Polytechnica. arXiv:2110.09815v1
42. Noguchi S, Inoue J (2021) Stochastic characterization and reconstruction of material microstructures for establishment of process-structure-property linkage using the deep generative model. Phys Rev E 104(2):025302
43. Bostanabad R, Chen W, Apley D (2016) Characterization and reconstruction of 3D stochastic microstructures via supervised learning. J Micros 264(3):282. https://doi.org/10.1111/jmi.12441
44. Cang R, Xu Y, Chen S, Liu Y, Jiao Y, Ren M.Y (2017) Microstructure representation and reconstruction of heterogeneous materials via deep belief network for computational material design. arXiv:1612.07401 [cond-mat, stat] pp 1–29
45. Faraji Niri M, Mafeni Mase J, Marco (2022) Performance evaluation of convolutional auto encoders for the reconstruction of li-ion battery electrode microstructure. Energies 15(12):4489. https://doi.org/10.3390/en15124489
46. Mosser L, Dubrule O, Blunt MJ (2017) Reconstruction of three-dimensional porous media using generative adversarial neural networks. Phys Rev E 96(4):1. https://doi.org/10.1103/PhysRevE.96.043309
47. Li X, Yang Z, Brinson LC, Choudhary A, Agrawal A, Chen W (2018) A deep adversarial learning methodology for designing microstructural material systems. In: Vol 2B: 44th design automation conference. American Society of Mechanical Engineers, Quebec City, Quebec, Canada, pp 1–14. https://doi.org/10.1115/DETC2018-85633
48. Iyer A, Dey B, Dasgupta A, Chen W, Chakraborty A (2019) A conditional generative model for predicting material microstructures from processing methods, arXiv:1910.02133 [cond-mat, stat]
49. Feng J, He X, Teng Q, Ren C, Honggang C, Li Y (2019) Reconstruction of porous media from extremely limited information using conditional generative adversarial networks. Phys Rev E 100:033308. https://doi.org/10.13140/RG.2.2.32567.98727
50. Fokina D, Muravleva E, Ovchinnikov G, Oseledets I (2020) Microstructure synthesis using style-based generative adversarial networks. Phys Rev E 101(4):1. https://doi.org/10.1103/PhysRevE.101.043308
51. Li Y, He X, Zhu W, Kwak H (2022) Digital rock reconstruction using Wasserstein GANs with gradient penalty, IPTC. In: International petroleum technology conference, p D012S123R001
52. Lee JW, Goo NH, Park WB, Pyo M, Sohn KS (2021) Virtual microstructure design for steels using generative adversarial networks. Eng Rep. https://doi.org/10.1002/eng2.12274
53. Amiri H, Vasconcelos I, Jiao Y, Chen PE, Plümper O (2022) Quantifying complex microstructures of earth materials: reconstructing higher-order spatial correlations using deep generative adversarial networks. Reconstructing higher-order spatial correlations using deep generative adversarial networks. preprint, Geology, Quantifying complex microstructures of earth materials. https://doi.org/10.1002/essoar.10510988.1
54. Hsu T, Epting WK, Kim H, Abernathy HW, Hackett GA, Rollett AD, Salvador PA, Holm EA (2020) Microstructure generation via generative adversarial network for heterogeneous, topologically complex 3D materials. arXiv:2006.13886 [cond-mat]
55. Henkes A, Wessels H (2022) Three-dimensional microstructure generation using generative adversarial neural networks in the context of continuum micromechanics. Comput Methods Appl Mech Eng 400:115497
56. Coiffier G, Renard P, Lefebvre S (2020) 3D geological image synthesis from 2D examples using generative adversarial networks. Front Water 2:560598
57. Kench S, Cooper SJ (2021) Generating 3D structures from a 2D slice with GAN-based dimensionality expansion. Nat Mach Intell 3:299. https://doi.org/10.1038/s42256-021-00322-1
58. Zhang F, He X, Teng Q, Wu X, Cui J, Dong X (2023) PM-ARNN: 2D-TO-3D reconstruction paradigm for microstructure of porous media via adversarial recurrent neural network. Knowl-Based Syst 264:110333
59. Shams R, Masihi M, Boozarjomehry RB, Blunt MJ (2020) Coupled generative adversarial and auto-encoder neural networks to reconstruct three-dimensional multi-scale porous media. J Petrol Sci Eng 186:1. https://doi.org/10.1016/j.petrol.2019.106794
60. Feng J, Teng Q, Li B, He X, Chen H, Li Y (2020) An end-to-end three-dimensional reconstruction framework of porous media from a single two-dimensional image based on deep learning. Comput Methods Appl Mech Eng 368:113043. https://doi.org/10.1016/j.cma.2020.113043
61. Zhang F, Teng Q, Chen H, He X, Dong X (2021) Slice-to-voxel stochastic reconstructions on porous media with hybrid deep generative model. Comput Mater Sci 186:110018. https://doi.org/10.1016/j.commatsci.2020.110018
62. Zhang Y, Seibert P, Otto A, Raßloff A, Ambati M, Kastner M (2023) DA-VEGAN: differentially augmenting VAE-GAN for microstructure reconstruction from extremely small data sets, arXiv:0904.3664 [cs]
63. Chamani H, Rabbani A, Russell KP, Zydney AL, Gomez ED, Hattrick-Simpers J, Werber JR (2023) Rapid reconstruction of 3-D membrane pore structure using a single 2-D Micrograph, arXiv preprint arXiv:2301.10601
64. Zhang T, Xia P, Lu F (2022) 3D stochastic reconstruction of porous media based on attention mechanisms and residual networks. Stoch Environ Res Risk Assess 36(4):1063

65. Zheng Q, Zhang D (2022) RockGPT: reconstructing three-dimensional digital rocks from single two-dimensional slice with deep learning. Comput Geosci 26(3):677
66. Phan J, Ruspini L, Kiss G, Lindseth F (2022) Size-invariant 3D generation from a single 2D rock image. J Petrol Sci Eng 215:110648
67. Düreth C, Seibert P, Rücker D, Handford S, Kästner M, Gude M (2023) Conditional diffusion-based microstructure reconstruction. Mater Today Commun 35:105608. https://doi.org/10.1016/j.mtcomm.2023.105608
68. Lee KH, Yun GJ (2023) Microstructure reconstruction using diffusion-based generative models. Mech Adv Mater Struct. https://doi.org/10.1080/15376494.2023.2198528
69. Lim HJ, Lee KH, Yun GJ (2023) Microstructure design of multifunctional particulate composite materials using conditional diffusion models. arXiv:2301.09051
70. Vlassis NN, Sun W (2023) Denoising diffusion algorithm for inverse design of microstructures with fine-tuned nonlinear material properties. Comput Methods Appl Mech Eng 413:116126
71. Li X, Zhang Y, Zhao H, Burkhart C, Brinson LC, Chen W (2018) A transfer learning approach for microstructure reconstruction and structure-property predictions. Sci Rep 8(1):13461. https://doi.org/10.1038/s41598-018-31571-7
72. Robertson AE, Kelly C, Buzzy M, Kalidindi SR (2023) Local-Global decompositions for conditional microstructure generation, local-global decompositions for conditional microstructure generation. preprint, SSRN. https://doi.org/10.2139/ssrn.4388214
73. Torquato S (2002) Random heterogeneous materials. Interdisciplinary applied mathematics, vol 16. Springer, New York. https://doi.org/10.1007/978-1-4757-6355-3
74. Jiao Y, Stillinger FH, Torquato S (2007) Modeling heterogeneous materials via two-point correlation functions: basic principles. Phys Rev E 76(3):1. https://doi.org/10.1103/PhysRevE.76.031110
75. Jiao Y, Stillinger FH, Torquato S (2008) Modeling heterogeneous materials via two-point correlation functions. II. Algorithmic details and applications. Phys Rev E 77(3):1. https://doi.org/10.1103/PhysRevE.77.031135
76. Feng J, Teng Q, He X, Qing L, Li Y (2018) Reconstruction of three-dimensional heterogeneous media from a single two-dimensional section via co-occurrence correlation function. Comput Mater Sci 144:181
77. Piasecki R, Olchawa W (2018) Statistical reconstruction of microstructures using entropic descriptors. Trans Porous Media 125(1):105. https://doi.org/10.1007/s11242-018-1012-7
78. Gerke KM, Karsanina MV, Vasilyev RV, Mallants D (2014) Improving pattern reconstruction using directional correlation functions. EPL (Europhysics Letters) 106(6):66002
79. Seibert P, Raßloff A, Kalina K, Safi A, Reck P, Peterseim D, Kästner M (2023) On the relevance of descriptor fidelity in microstructure reconstruction. PAMM 23(3):e202300116
80. Adam A, Wang F, Li X (2022) Efficient reconstruction and validation of heterogeneous microstructures for energy applications. Int J Energy Res 85:8578. https://doi.org/10.1002/er.8578
81. Alexander SK, Fieguth P, Ioannidis MA, Vrscay ER (2009) Hierarchical annealing for synthesis of binary images. Math Geosci 41(4):357. https://doi.org/10.1007/s11004-008-9209-x
82. Pant LM, Mitra SK, Secanell M (2015) Multigrid hierarchical simulated annealing method for reconstructing heterogeneous media. Phys Rev E 92(6):063303
83. Karsanina MV, Gerke KM (2018) Hierarchical optimization: fast and robust multiscale stochastic reconstructions with rescaled correlation functions. Phys Rev Lett 121(26):265501
84. Chen D, Xu Z, Wang X, He H, Du Z, Nan J (2022) Fast reconstruction of multiphase microstructures based on statistical descriptors. Phys Rev E 105(5):055301
85. Seibert P, Raßloff A, Kalina KA, Gussone J, Bugelnig K, Diehl M, Kästner M (2023) Two-stage 2D-to-3D reconstruction of realistic microstructures: implementation and numerical validation by effective properties. Comput Methods Appl Mech Eng 412:116098
86. Pant LM, Mitra SK, Secanell M (2014) Stochastic reconstruction using multiple correlation functions with different-phase-neighbor-based pixel selection. Phys Rev E 90(2):1. https://doi.org/10.1103/PhysRevE.90.023306
87. Rozman MG, Utz M (2001) Efficient reconstruction of multiphase morphologies from correlation functions. Phys Rev E 63(6):1. https://doi.org/10.1103/PhysRevE.63.066701
88. Xu H, Dikin DA, Burkhart C, Chen W (2014) Descriptor-based methodology for statistical characterization and 3D reconstruction of microstructural materials. Comput Mater Sci 85:206. https://doi.org/10.1016/j.commatsci.2013.12.046
89. Scheunemann L, Balzani D, Brands D, Schröder J (2015) Design of 3D statistically similar representative volume elements based on Minkowski functionals. Mech Mater 90:185. https://doi.org/10.1016/j.mechmat.2015.03.005
90. Seibert P, Husert M, Wollner MP, Kalina KA (2024) Kästner M (2023) Fast reconstruction of microstructures with ellipsoidal inclusions using analytical descriptors. Comput-Aided Design 166:103635
91. Groeber M (2008) A framework for automated analysis and simulation of 3D polycrystalline microstructures. Part 1: statistical characterization. Acta Materialia 56(6):1257
92. Quey R, Kasemer M (2022) The Neper/FEPX project: free / open-source polycrystal generation, deformation simulation, and post-processing. IOP Conf Ser Mater Sci Eng 1249(1):012021
93. Prasad M, Vajragupta N, Hartmaier A (2019) Kanapy: a Python package for generating complex synthetic polycrystalline microstructures. J Open Source Softw 4(43):1732. https://doi.org/10.21105/joss.01732
94. Seibert P, Ambati M, Raßloff A, Kästner M (2021) Reconstructing random heterogeneous media through differentiable optimization. Comput Mater Sci 196:110455
95. Seibert P, Raßloff A, Ambati M, Kästner M (2022) Descriptor-based reconstruction of three-dimensional microstructures through gradient-based optimization. Acta Materialia 227:117667
96. Bostanabad R (2020) Reconstruction of 3D microstructures from 2D images via transfer learning. Comput-Aided Design 128:102906. https://doi.org/10.1016/j.cad.2020.102906
97. Bhaduri A, Gupta A, Olivier A, Graham-Brady L (2021) An efficient optimization based microstructure reconstruction approach with multiple loss functions. Comput Mater Sci 199:110709
98. Zhou XP, Xiao N (2018) 3D numerical reconstruction of porous sandstone using improved simulated annealing algorithms. Rock Mech Rock Eng 51(7):2135
99. Xiao N, Zhou X, Berto F (2021) Fracture analysis of rock reconstruction models based on cooling-solidification annealing algorithms. Fatigue Fract Eng Mater Struct 44(9):2503–2523
100. Talukdar M, Torsaeter O, Ioannidis M, Howard J (2002) Stochastic reconstruction, 3D characterization and network modeling of chalk. J Petrol Sci Eng 35(1–2):1. https://doi.org/10.1016/S0920-4105(02)00160-2
101. Gerke KM, Karsanina MV, Skvortsova EB (2012) Description and reconstruction of the soil pore space using correlation functions. Eurasian Soil Sci 45(9):861
102. Seibert P, Raßloff A, Kalina K, Ambati M, Kästner M (2022) Microstructure characterization and reconstruction in Python: MCRpy. Integr Mater Manuf Innov 11(3):450

103. Robertson AE, Kalidindi SR (2021) Efficient generation of anisotropic N-field microstructures from 2-point statistics using multi-output Gaussian random fields. SSRN Electron J 232:117927
104. Jiang Z, Chen W, Burkhart C (2013) Efficient 3D porous microstructure reconstruction via Gaussian random field and hybrid optimization. J Microsc 252(2):135. https://doi.org/10.1111/jmi.12077
105. Zhang H, Yu H, Meng S, Huang M, Micheal M, Su J, Liu H, Wu H (2022) Fast and accurate reconstruction of large-scale 3D porous media using deep learning. J Petrol Sci Eng 217:110937
106. Su J, Xiao G, Zhang H (2022) Microstructure reconstruction of porous copper foams based on deep convolutional generative adversarial networks with physical characteristics of materials. Int J Data Sci Anal 14(1):89–97
107. Li Y, Jian P, Han G (2022) Cascaded progressive generative adversarial networks for reconstructing three-dimensional gray-scale core images from a single two-dimensional image. Front Phys 10:716708
108. Yang Z, Li X, Catherine Brinson L, Choudhary AN, Chen W, Agrawal A (2018) Microstructural materials design via deep adversarial learning methodology. J Mech Design 140(11):1. https://doi.org/10.1115/1.4041371
109. Henrich M, Fehlemann N, Bexter F, Neite M, Kong L, Shen F, Könemann M, Dölz M, Münstermann S (2023) DRAGen–A deep learning supported RVE generator framework for complex microstructure models. Heliyon 9(8):e19003
110. Macedo RBd, Monfared S, Karapiperis K, Andrade JE (2023) What is shape? Characterizing particle morphology with genetic algorithms and deep generative models. Granul Matter 25(1):2. https://doi.org/10.1007/s10035-022-01282-y
111. Mordvintsev A, Randazzo E, Niklasson E, Levin M (2020) Growing neural cellular automata. Distill 5(2):23. https://doi.org/10.23915/distill.00023
112. Mordvintsev A, Niklasson E, Randazzo E (2021) Texture generation with neural cellular automata. arXiv:2105.07299
113. Gatys L, Ecker AS, Bethge M (2015) Texture synthesis using convolutional neural networks. arXiv:1505.07376 pp. 1–9
114. Reck P, Seibert P, Raßloff A, Peterseim D, Kästner M (2023) Scattering transform in microstructure reconstruction. PAMM 23(3):e202300169
115. Efros AA, Freeman WT (2001) Image quilting for texture synthesis and transfer. In: Proceedings of the 28th annual conference on computer graphics and interactive techniques–SIGGRAPH '01. (ACM Press, Not Known), pp 341–346. https://doi.org/10.1145/383259.383296
116. Liang L, Liu C, Xu YQ, Guo B, Shum HY (2001) Real-time texture synthesis by patch-based sampling. ACM Trans Graph 20(3):127. https://doi.org/10.1145/501786.501787
117. De Bonet JS (1997) Multiresolution sampling procedure for analysis and synthesis of texture images. In: Proceedings of the 24th annual conference on computer graphics and interactive techniques – SIGGRAPH '97 (ACM Press, Not Known), pp 361–368. https://doi.org/10.1145/258734.258882
118. Wei LY, Levoy M (2000) Fast texture synthesis using tree-structured vector quantization. In: Proceedings of the 27th annual conference on computer graphics and interactive techniques - SIGGRAPH '00 (ACM Press, Not Known), pp 479–488. https://doi.org/10.1145/344779.345009
119. Commons C (2021) Creative cmmons licence CC BY 4.0, ceative cmmons licence CC BY 4.0. https://creativecommons.org/licenses/by/4.0/legalcode